DOCUMENT RESUME

ED 207 550 IR 009 661

AUTHOR Mayer, Richard E.; Bayman, Piraye
TITLE Analysis of Users' Intuitions About the Operation of Electronic Calculators. Report No. 80-4. Series in Learning and Cognition.
INSTITUTION California Univ., Santa Barbara. Dept. of Psychology.
SPONS AGENCY National Inst. of Education (DHEW), Washington, D.C.
PUB DATE 80
GRANT NIE-G-78-0162; NIE-G-80-0118
NOTE 78p.

EDRS PRICE MF01/PC04 Plus Postage.
DESCRIPTORS *Calculators; *Cognitive Style; *Computation; Higher Education; *Learning Processes; Psychological Studies; Statistical Analysis; Tables (Data); Undergraduate Students
IDENTIFIERS Computer Literacy; *Intuition (Mathematics)

ABSTRACT

A study designed to serve as an initial step in building a theory of computer literacy was conducted to provide new information concerning how humans think about calculators and to determine how individual differences in students' intuitions affect their understanding and use of the calculator. Thirty-three expert and 33 novice calculator users were asked to fill out questionnaires, and then to predict what number would be in the display of the calculator after a series of key presses for 88 math problems. Analytic techniques of cognitive psychology were applied to assess the performance of the subjects compared to the performance of the calculators, as well as subject performance in light of the type of calculator used, and the amount of previous experience with calculators. Results indicated that there were tremendous individual differences among users in their interpretations of the logic of the calculator's operating system. Future work is recommended to determine whether intuitions, once diagnosed, can be altered through instruction, and to determine whether people with certain intuitions can use their calculators more creatively c learn a new computer language more efficiently than people with other sets of intuition. Nineteen references are listed and supporting data are appended. (MER)

ED207550

SERIES IN LEARNING AND COGNITION

Analysis of Users' Intuitions About the Operation of Electronic Calculators

Richard E. Mayer and Piraye Bayman

Report No. 80-4

This research was supported by Grant NIE-G-78-0162 from the National Institute of Education and Grant NIE-G-80-0118 from the National Institute of Education.

R009661

Abstract

Thirty-three novice and 33 expert users predicted what number would be in the display of a calculator after a sequence of key presses (such as 2 + 3 +). The performance of each subject on 88 problems was formally described as a 13 line production system. Conditions were key presses (such as + after a number); actions were changes in the display or internal registers (such as display the evaluation of the expression in the register). Large individual differences were observed. Differences among subjects included when an expression is evaluated and displayed (e.g., after a + key, x key, = key, and/or number key), whether or not the display is incremented when two operation keys were pressed in sequence (e.g., 2 + + or 2 x x), whether or not the display is incremented when an equals was pressed after an operation (e.g., 2 + =), what the order of arithmetic would be in a chain (e.g., 2 + 3 x 7). Experts were more consistent in their performance and tended to be more likely than novices to base answers on standard operating systems. Implications for developing a theory of computer literacy were discussed.

Analysis of Students' Intuitions About the Operation of Electronic Calcuators

Within the past decade electronic calculators have become a part of our society, including widespread and rapid acceptance in schools (Mullish, 1976). Based on a survey of articles and editorials published within the past few years in Arithmetic Teacher and Mathematics Teacher, as well as a policy statement by the National Council of Teachers of Mathematics (1976), it is clear that calculators will play an important role in the education of our children. For example, the following statements by mathematics educators are typical; "All students will profit from having access to a calculator" (Gawronski & Coblentz, 1976). "I propose that we make fullest possible use of calculators in all grades of our school" (Hopkins, 1976). "Not since the printing press has any invention had such potential for revoluntionizing education, particularly mathematics education" (Rudnick & Krulik, 1976).

However, in spite of these optimistic predictions and endorsements, the research community has been slow in providing information that might be useful in this impending calculator-curriculum revolution. For example, most experimental studies to date have compared changes in achievement and/or attitude scores for students who use calculators in school versus students who were not allowed to use calculators (Gastin, 1975; Roberts & Fabrey, 1978; Roberts & Glynn, 1979; Schnur & Land, 1976; Suydam, Note 1). In a recent review of 34 studies, most of which were not published in journals, Roberts (1980) observed that there was clear evidence that calculators improve computational efficiency but no concensus concerning effects on higher level conceptual achievement or attitudes towards mathematics. Thus, he concludes that "the research literature offers no guidance" concerning how to incorporate calculators into school curricula.

The present paper does not attempt to address the question of whether calculators influence changes in mathematics achievement and attitude scores. Rather, this paper is based on the idea that since calculators will become a part of everyday life for students, it is important to know how people come to understand and interact with calculators. The calculator represents a student's first introduction to a computer, to a computer language, and to computer literacy in general. Pressing a key is analogous to a computer command. In spite of tremendous breakthroughs in development of improved calculator hardware for the mass market, there has been comparatively little work on what Shneiderman (1980) calls "software psychology". That is to say, we know very little about how people understand calculators, what types of instruction will help people become creative users of calculators, why some people seem to not use them very well, or how to design operating systems that make psychological sense.

Since calculator useage seems so simple and since even children can teach themselves to use a calculator in a short time, some educators have suggested that explicit instruction or concern about users' understanding of calculators is not needed (Bell, 1976). This might be correct if one's goal is simply to have students use the calculator as a "black box" that gives answers for mundane computation. However, when the goal is to promote productive problem solvers, there is reason to believe that the student's understanding of how the calculator operates is important. For example, Scandura, Lowere, Veneski & Scandura (1976) found that some students who are left to teach themselves develop bizzare intuitions; for example, some subjects concluded that the plus (+) and equals (=) key did nothing since they caused no visible change in the display. On the other hand, Mayer (1980) found that by letting fourth-graders explore the functions of the operation keys some of them discovered that pushing the same operation key more than once would cause the calculators to repeat the process with the number last entered. Leaving the understanding of the basic functions of the calculator to chance might create as many students with incorrect intuitions as correct ones. Thus, although two users may be able to use the calculator to solve basic computational problems, there may be large individual differences in the way users understand and interact with calculators.

Elucidation of individual differences in students' intuitions about the operation of calculators (i.e., students' conception of what goes on inside the "black box" when a key is pressed) is the logical first step in building a theory of computer literacy, and is the goal of the present study. In particular, this study addresses two related issues.

(1) What knowledge do people have about how calculators work? Since most users are "self-taught" and seem to be able to use their calculators, an important issue concerns what they have learned. Since some intuitions may lead to more creative use of calculators and to better transfer to computer languages (such as programmable calculators or BASIC), it would be useful to be able to describe exactly what people's intuitions are. Recent research on the cognitive analysis of computational skill suggests that two children with the same apparent performance may be using entirely different conceptions of computation. For example, Groen & Parkman (1972) have developed cognitive models of addition, and Woods, Resnick & Groen (1975) have developed models of subtraction. More recently, Brown & Burton (1975) have developed a BUGGY program which serves to diagnose problems in a child's arithmetic procedures by developing a formal description of the algorithm that the child is using. Successful application of cognitive analysis tools to the problem of describing computational skill encourages the idea that similar techniques can be used to formalize how students understand calculator logic.

(2) What are the differences in intuitions among individuals? For example, do "experts" have different intuitions than "novices?" In recent studies, Larkin (1979) and Simon & Simon (1978) have been able to formalize differences between what experts and novices know about solving physics problems. Larkin (1979) has been able to represent the differences in terms of differences in the organization and size of productions in a production system. Similar techniques may be applied to representing differences between experts and novices in the domain of calculator use.

STUDIES 1 AND 2

The purpose of study 1 was to determine the knowledge that ordinary users have concerning the operation of hand-held calculators. In particular, the goal was to formally describe each subject's conception of the calculator's operating system. The purpose of study 2 was to determine whether the formal descriptions developed in study 1 would also describe the conceptions of people who were more knowledgeable about operating systems. For purposes of this paper, subjects in study 1 are labeled "novices" and subjects in study 2 are labeled "experts".

Method

Subjects

The subjects in study 1 were 46 college undergraduates recruited from the subject pools at the University of Pittsburgh and the University of California, Santa Barbara.[1] All subjects participated in order to fulfill a requirement for their introductory psychology courses. Subjects in study 1 had no previous experience with computer programming nor with the concept of operating systems. Thirteen of the 46 subjects in study 1 produced inconsistent performance, so only data for the remaining 33 subjects was used for the analyses. Of these 33 subjects there were 16 females and 17 males, and 26 of the subjects owned a calculator. The mean age was 18 years, the mean GPA was 3.0, the mean SAT-Quantitative score was 547, the mean SAT-Verbal score was 491.

The subjects in study 2 were 35 college undergraduates recruited from a course in computer programming at the University of California, Santa Barbara.[2] All subjects had taken at least one previous course in computer programming and were currently in a course that included study of operating systems. Of these 35 subjects, one gave inconsistent performance and one failed to follow directions. The remaining 33 subjects were retained for analysis in this study. There were 13 females and 20 males, and 32 of the subjects owned a calculator. The mean age was 21 years, the mean GPA was 2.9, the mean SAT-Quantitative score was 669, and the mean SAT-Verbal score was 552.

The main difference between subjects in study 1 (novices) and study 2 (experts) is that all of the experts had formal instruction in computer programming and some introduction to operating systems while none of the novices did; the experts were older, $t(60) = 2.66$, $p < .01$; and the experts had higher SAT-Quantitative scores, $t(44) = 4.67$, $p < .001$. Thus, while the main focus was on comparing "liberal arts" students who had no formal programming experience to "engineering" students who had formal training in programming, any comparisons between the subjects in the two studies must be made in light of other differences such as age and SAT scores.

Materials

The materials in study 1 and study 2 were essentially identical. Materials consisted of a questionnaire, an instruction sheet, and two two-page problem sets.

The questionnaire was an 8½ x 11 inch sheet of paper which asked the subject to indicate his or her age, sex, GPA, SAT scores, year in school, major in school, experience with computer programming, experience with calculators, and previous mathematics courses. In particular, subjects were asked to indicate whether they owned or regularly used a particular calculator, and if so, to give the name of the model. In addition, subjects were asked to check a box corresponding to the average number of minutes per week they used a calculator--less than 10, 10 to 30, 30 to 60, more than 60.

The instructions for the problems were typed onto an $8\frac{1}{2}$ x 11 inch sheet of paper. Instructions described a typical calculator and the task.

Each of the two problem sets consisted of 44 problems typed onto two $8\frac{1}{2}$ x 11 inch sheets of paper with one problem per double-spaced line. Each problem presented a series of key strokes and provided a blank space for the subject to indicate what number would be in the display. Each problem contained from one to seven key strokes and each key stroke was either a single digit (2, 3 or 7), a plus key (+), a multiply key (x), or an equals key (=). The two problem sets (Set A and Set B) provided for a reliability check since each problem in Set A corresponded to a problem of the same form in Set B, and both sets presented the corresponding problems in the same order. However, the specific digits used in corresponding problem were different. For example, problems 2+3+7 or 2+3x or 2+=+=+= in set A corresponded to 7+3+2 or 7+3x or 7+=+=+=, respectively, in Set B. The complete list of Set A problems is given in the left side of Table 1.

Procedure

The procedures were essentially identical in study 1 and study 2 except that subjects were run individually in study 1 and were run as a group in study 2.

First, each subject filled out the questionnaire. Then the instructions were presented. Subjects were asked to suppose that they had just been given a new standard four-function calculator that worked efficiently, and to suppose that they would be using this calculator throughout the session. They were told that for each problem, their job was to predict what number would be in the display of the calculator after the series of key presses (assuming the calculator was cleared at the beginning of the problem). Then, one of the problem sets was randomly selected and given to the subject; when the subject finished the first set, the other set was given. Subjects were asked to put their answers in the space next to each problem and there was no time limit.

Results and Discussion

Scoring

The data for each subject in each study consisted of a number (i.e., the subject's answer) for each of the 88 problems.

Reliability of Performance

Since two forms of the same 44-problem test were administered to each subject, it was possible to determine the reliability of each subject's performance. For each of the possible answers to the 44 problems in set A, corresponding answers were generated for set B. For example, if a subject gave 12 as the answer for 2+3+7 in set A, the corresponding answer for 7+3+2 in set B would also be 12; if a subject gave 7 as the answer for the above problem in set A, the corresponding answer in set B would be 2. Similarly, if a subject gave 2 as the answer for 2+=+=+= in set A, the corresponding answer for 7+=+=+= in set B is 7; if a subject gave 16 for the above problem in set A, the corresponding answer for set B is 56. Reliability scoring was conducted by matching each of the 44 problems in set A with its corresponding problem in set B; if the answers did not correspond, subjects were given a point.

Thirty-three of the 46 subjects in study 1 displayed six or less (i.e., less than 14%) non-matching scores between set A and set B. Data for the 13 subjects who displayed more than six non-matches (i.e., over 14% unreliable answers) were not analyzed further. The mean non-matching score for all 46 subjects in study 1 was 5.3 or 12%; the mean non-matching score for the 33 selected subjects in study 1 was 2.6 or 6%.

In study 2, one of the 35 subjects gave more than six non-matching answers between set A and set B. Data for this subject as well as for one subject who failed to follow directions were not included in subsequent analyses. The mean number of non-matching answers for the 33 selected subjects in study 2 was .8 or 2%.

One question that may be raised at this point is whether the experts and novices differ with respect to the reliability of their performances. The proportion of unreliable novices (13 out of 46) was significantly higher than the proportion of unreliable experts (1 out of 35) as determined by a chi-square test, $\chi^2 = 7.28$, df = 1, $p < .01$. In addition, for the 33 selected subjects in each study, the novices produced significantly more unreliable answers than the experts as determined by a t-test, $t(64) = 10.59$, $p < .001$. Thus, as might be expected, experts were more consistent in the way they answered problems than were the novices.

All subsequent analyses are based on answers to set A for the 33 subjects in each group.[3]

Performance of Subjects Compared to Performance of Calculators

In this section we address the question of which calculator most closely fits the answers given by the subjects. Of the 33 subjects in study 1, 17 owned simple Texas Instruments (TI) models, three owned Sharp models, one owned a Rockwell model, one owned a Hewlett-Packard HP-21, and eleven either did not own a calculator or could not remember what kind they owned.

Since TI, Sharp and Rockwell were the relevant models[4] owned by subjects in study 1, answers to each of the 44 problems in set A were generated using each of the three brands of calculator. Interestingly, while most of the calculators gave identical answers for most problems, there were different answers produced by at least two of the calculators on 20 of the 44 problems.

A difference score was computed for each subject for each of the three calculator brands. The difference score was based on tallying the number of times that the subject's answer was not identical to the calculator's answer for the 44 problems in set A. Mean difference scores in study 1 were 8.8 (20%) for TI, 20.8 (47%) for Rockwell, and 14.9 (34%) for Sharp. A one-way analysis of variance was conducted on the difference scores with brand of calculator as a within subjects factor. The ANOVA revealed that the difference scores listed above were significantly different from one another, $F\ (2,64) = 42.6$; $p < .001$. Supplementary Newman-Keuls tests indicated that the score for TI was significantly better than for Rockwell, but no other differences were significant ($p < .05$).

Of the 33 subjects in study 2, 23 owned Texas Instruments (TI) models, two owned Sharp models, three owned Casio models, three owned Hewlett-Packard, and two either did not own a calculator or could not remember what kind they owned. Thus, as with novices the most freqpuntly owned calculator was TI. However, 11 of the 33 novices did not own or could not remember which calculator they owned, while only 2 of the 33 experts fell into this category. According to a chi-square test, this difference between proportions is significant, $\chi^2 = 6.13$, df = 1, $p < .025$.

The mean difference scores in study 2 were 90 (20%) for TI, 18.2 (41%) for Rockwell, and 13.7 (31%) for Sharp. A one-way analysis of variance was conducted on the difference scores with brand of calculator as a within subjects factor. The ANOVA revealed that the difference scores listed above were significantly different from one another, $F\ (2,64) = 23.74$, $p < .001$. Supplementary Newman-Kuels tests revealed that the score for TI was significantly better than for Rockwell or for Sharp, and that the score for Sharp was significantly better than for Rockwell ($p < .05$).

Thus, both in study 1 and in study 2, there is evidence that TI's operating system most closely matches the intuitions of subjects for the 44 problems in the test. In comparing experts and novices, there is no evidence of any differences in which calculator gives the best fit; t-tests revealed no differences between experts and novices with respect to their scores for TI, $t(64) < 1$, for Rockwell, $t(64) = 1.58$, or for Sharp, $t(64) = 1.27$.

Performance of Subjects By Type of Calculator They Own

The previous section suggested that subjects in our sample generated performance that more closely matched the performance of a TI calculator than the other calculators we tested. However, since the TI is the brand of calculator that most subjects in our sample owned, the above results may be mainly due to experience with TI calculators. In order to test this idea, subjects in study 1 were divided into two groups: those who owned a TI calculator (n = 17) and those who do not (n = 16).

The mean difference scores for the TI-owners in study 1 were : TI =8.0, Rockwell = 20.0, and Sharp = 14.1; the mean difference scores for the non-TI owners were: TI = 9.8, Rockwell = 21.6, Sharp = 15.8. As can be seen, for both TI-owners and non TI-owners, the difference scores are least for TI. An analysis of variance was conducted on the difference score data with group as a between subjects factor and type of calculator as a within subjects factor.

There was a significant difference between the calculators in how well they matched the performance of all subjects, F (2,60) = 51.16, $p < .001$, but there was no group x calculator interaction, F (4,60 < 1. Thus, there was no evidence that the non-TI owners were different from TI owners with respect to their performance being best fit by a TI calculator. A separate one-way ANOVA was conducted on the data for the non-owners with type of calculator as a within-subjects factor. The differences in difference scores were significant, F (2,30) = 37.57, $p < .001$; subsequent Newman-Kuels tests showed that the score for the TI were significantly better than for Sharp or Rockwell among non-TI owners ($p < .05$).

As an additional analysis, the performance of each subject was labeled as TI-like, Rockwell-like or Sharp-like, based on which of the three calculators produced the lowest difference score for the subject. Of the 33 subjects in study 1, 29 were classified as TI-like and 4 were better matched with other brands. For the TI-owners, 15 were classified as TI-like and 2 were best fit by other brands; for the non-TI owners, 14 were classified as TI-like and 2 were best fit by other brands. A Fisher's Exact test revealed that there was no evidence of any differences among the two groups (TI owners versus non-owners) in the proportion of TI-like [illegible]s.

The performance of the experts in study 2, like the novices in study 1, most closely matches the performance of the TI, but this may be due to the fact that TI is the most prevelant calculator used among the experts. As in study 1, this idea was tested by dividing the experts into those who owned a TI calculator (n = 22) and those who did not (n = 11). The mean difference scores for the TI-owners were: TI = 76, Rockwell = 34.4, Sharp = 13.2; the main difference scores for the non-TI owners were: TI = 11.8, Rockwell = 17.4, Sharp = 14.6. An analysis of variance was conducted on the difference score data with ownership group as a between subjects factor and type of calculator as a within subjects factor. As expected, there was a significant difference among the calculators in how well they matched the performance of all subjects, $\underline{F}$ (2,60) = 19.21, $\underline{p} < .01$; however, as in study 1, there was no interaction between group and calculator, $\underline{F}$ (4,60) < 1. Thus, as in study 1, there was no evidence that non-TI owners were different from TI owners with respect to their performance being best fit by a TI calculator.

As an additional analysis, the performance of each subject in study 2 was labeled as TI-like, Rockwell like, or Sharp-like, based on which of the three calculators produced the lowest difference scores for each subject. Of the 33 subjects, 26 were classified as TI-like and 7 were better matched by other brands. For TI-owners, 18 were classified as TI-like and 4 were best fit by other brands; for the non-TI owners, 8 were classified as TI-like and 3 were best fit by other brands. A Fisher's Exact test revealed that there was no evidence of any differences among the two groups (TI owners versus non-owners) in the proportion of TI-like subjects.

This section helps clarify the earlier finding that subjects' performance is most closely matched by TI's operating system. Since this finding seems to be present for both TI-owners and non-owners it may be attributed to the "intuitive appeal" of the TI operating system rather than to users having more experience with TI products. However, it should be pointed out that the correspondence between the calculator's answers and the subjects' answers are far from perfect, even when we choose the best fitting calculator.

<u>Performance of Subjects by Amount of Experience With Calculators</u>

This section explores the issue of whether subjects who differ with respect to how much they use a calculator also differ with respect to intuitions about the operating system underlying the calculator. In order to address this issue, subjects in study 1 were divided into two groups based on their reported weekly use of calculators: low—less than 10 minutes per week (n = 17), and moderate--10 or more minutes per week (n = 16).

The mean difference scores for study 1 on each of the three calculators brands was TI = 10.2, Rockwell = 22.2, and Sharp = 16.5 for the low experience group, and TI = 7.4, Rockwell = 19.3, and Sharp = 13.3 for the moderate experience group. As can be seen, TI produces the lowest score (i.e., best fit) for subjects' performance in both groups. An analysis of variance was conducted on the difference score data with experience (low vs. moderate) as a between subjects factor and calculator brand as a within subjects factor. As in previous analyses, there was a significant overall difference among the calculators in how closely they fit the intuitions of the subjects, $\underline{F}$ (2,62) = 40.62, $\underline{p}$ < .001, and there was no interaction between group and calculator, $\underline{F}$ (2,62) < 1. Thus, there was no evidence in study 1 that the superior fit of the TI calculator was influenced by how much experience a subject had.

As in the previous section, each subject was classified as being either TI-like, Rockwell-like, or Sharp-like based on which calculator produced the least number of differences with the subject's actual performance. For the low experience group in study 1, 15 subjects were best fit by TI and 2 were best fit by another calculator; for the moderate experience group 14 were best fit by TI and 2 were best fit by another calculator. A Fisher's Exact test showed that there were no significant differences between low experience and moderate experience groups with respect to the proportion of TI-like subjects. Thus, there is no evidence in this analysis that amount of experience with calculators influence the subjects' intuitions about the operating systems of calculators.

In study 2, eight subjects were in the low experience category and 25 indicated moderate experience. Corresponding analyses were conducted on the data for the experts in study 2. The mean difference scores for study 2 were: TI = 9.5, Rockwell = 18.1, and Sharp = 14.4 for the low experience group, and TI = 8.8, Rockwell = 18.3, and Sharp = 13.5 for the moderate experience group. An ANOVA revealed the expected overall effect, $F\ (2,52) = 18.92$, $p < .001$; but there was no interaction between experience group and calculator, $F\ (2,62) < 1$, and thus no evidence that the superior fit of the TI calculator was influenced by how much experience a subject had with calculators. For the low experience group in study 2, 6 of the 8 subjects were best fit by TI while for the moderate experience group, 21 of the 25 subjects were best fit by TI. A Fisher's Exact test shows there is no significant difference with respect to the proportion of TI-like subjects among low and moderate experience subjects in study 2. Thus, for both experts and novices, there is no evidence that experience with calculators influences the subjects' intuitions about the operating system of calculators.

Inter-Subject Consistency in Performance

The foregoing analyses indicates that subjects' performance was closest to that of a TI calculator, and that this pattern was not influenced by whether subjects actually owned a TI calculator nor whether they had experience with using a calculator. However, the foregoing analyses also made clear that subjects' performance could not be adequately described as corresponding to one particular calculator's performance, since the best fitting calculator predicted only about 80% of the answers. In this section, we explore the question of how similar or different the subjects' answers were from subject to subject.

Table 1 gives a list of the 44 problems in set A as well as the answers given by subjects in study 1 and study 2. Each answer that was given by any subject is listed (in parentheses) along with the number of subjects who gave that answer in study 1 and in study 2. As can be seen, for each question there is an answer that occurs most often (i.e., the modal answer) and there may be one or more other answers given by some subjects (i.e., alternative answers). The percentage of subjects' answers that are modal answers, i.e., answers that correspond to the most common answer for each question is 83% for study 1 and 81% for study 2. A t-test revealed that the experts and novices do not differ with respect to the percentage of modal answers, $t(64) < 1$. Kolmogorov-Smirnov's One-Sample tests based on the data for both studies together indicated that the following problems produce significant ($\underline{P} < .05$) number of non-model answers: 4, 13, 15, 17, 19, 21, 23, 24, 25, 27, 28, 29 and 31 through 44. Thus, the data in Table 1 encourages the conclusion that there are substantial individual differences among subjects in their intuitions about calculators. In order to more closely examine and describe these differences, all subsequent analyses will involve descriptions of single subjects rather than group data.

Insert Table 1 about here

Analysis of Performance of Individual Subjects on Simple-Problems

The goal of the analysis in this section is to provide a formal description of the knowledge that each subject has concerning how the calculator solves simple problems. Thus, for each subject, a model was developed which could generate the subject's answers on simple test problems. This section describes the data source, the format of the models, how the data for a subject were fit by a particular model.

Data Source. This analysis was based on the data for the 33 novices and the 33 experts who gave consistent answers (i.e., high reliability between the two corresponding question sets). In order to provide an intensive analysis of the performance of each subject in the sample, this analysis focused on only the simple problems. Simple problems are defined as those which contain symbols for number and/or plus and/or equals but which do not contain any multiplication symbols. Eighteen of the 44 problems in set A fit this description; these are problems 1 through 10 and 27 through 34 in Table 1. Thus, for each of the 33 novices and 33 experts, the data source was a list of 18 answers for the 18 simple problems.

Development of Models. The goal of the present analysis was to develop simple production system models that would generate the performance of each subject on the 18 simple problems. Because of its efficiency and apparent relevance for the present task, a production system was used to represent the knowledge of each subject. A production system contains a list of productions with each production consisting of a condition and a corresponding action. Conditions, actions, and productions for the current problem are described in this section.

The relevant conditions for the present analysis are related to having pressed one of the keys on the calculator keyboard. The key relevant to the simple problems are the ten number keys (0, 1, 2, 3, 4, 5, 6, 7, 8, 9), the addition operation key (+) and the equals key (=). Table 2 shows nine conditions that could exist; three are not relevant to the simple problems since these conditions are never found in the problems. The other six conditions contain an exhaustive list of the conditions present in the simple problems. At first blush, it might seem that Table 2 gives many redundant conditions and that a simplier scheme is to deal with only three basic conditions--pressing a number (#), pressing a plus (+), or pressing an equals (=). However, the performance and comments of subjects suggest that a key press means something different to some people depending on the immediately preceeding key press.

Insert Table 2 about here

The actions for each production consist of events that take place inside the calculator. Following a system developed to represent the "internal" actions in computer languages (Mayer, 1979), each of the calculator actions can be represented as a triplet: some <u>operation</u> is applied to some <u>object</u> at some <u>location</u> in the calculator. The operations consist of the following:

(1) Create -- A number of expression is placed in the display or register, e.g., when you press a number key that number appears in the display.

(2) Destroy -- A number or expression is erased from the display or register, e.g., when you press the equals key the previous number in the display is erased (and replaced with a new one).

(3) Evaluation -- An expression (from the register) is converted into a single number, e.g., the evaluation of 3 + 2 is 5. (For the current example, evaluation of a number or a number followed by an operation is the number, e.g., evaluation of 3 is 3, evaluation of 2 + is 2).

The objects consist of:

(1) Numbers -- A number is any single or multiple digit sequence such as 2, 14, 156, etc.

(2) Operation -- An operation is a mathematical symbol for some arithmetic computation such as addition (+), or multiplication (x).

(3) Expression -- An expression is a sequence consisting of numbers and operations such as, 2 + 3 or 2 + or 2 + 3 x 4.

The locations consist of:

(1) Display -- The external display in a calculator normally consists of at least eight places, where each place can hold one digit.

(2) Register -- An internal register is inside the calculator and consists of subregisters for individual numbers and for operations. Expressions are held in the order of input, with the first number on the left, followed by the operation, followed by the next number.

Table 3 shows some typical actions that may occur for the simple problems. It should be noted that the 12 actions listed in the table actually refer to groups of single actions. For example, D = # consists of two single actions: erasing the old number from the display and replacing it with a new number. Also, it should be noted that the first five actions refer only to the display, and have no effect on changing the register; the other actions refer only to the register and have no effect on changing the display.

Insert Table 3 about here

Since Table 2 gives a list of all relevant conditions and Table 3 gives a list of relevant actions, it is now possible to describe any production as one of the conditions coupled with one of the actions. For example, the production,

P5. If # after + Then D = # and R = R + #

means that when the number key is pressed after the plus key the result is that the old number in the display is replaced with the new number, and the old expression in the register is retained but the new number is added to the right. Thus, if the sequence had been 5 + 2 + and now a 3 is pressed, the display will contain a 3 (before there was a 2) and the register will contain the expression 5 + 2 + 3 (before there was 5 + 2 +).

Fitting a model to each subject. The foregoing sections described that the data source for each subject was the 18 answers to the target questions and that models are based on (alternative) actions associated with each of the six conditions. Thus, for the data of each subject the goal is to develop a production system which consists of six productions.

The task was made somewhat easier by the fact that there are groups of subjects who gave identical answers for the simple problems. Based on their answers to the 18 simple problems, subjects could be grouped in one of six distinct categories where subjects in each of the first five groups produced identical answers with one another. Group 1 contained 8 novices and 11 experts, all of whom gave answers that were identical to answers produced by inexpensive TI models. Group 2 contained 10 novices and 7 experts, all of whom gave answers that were identical to Group 1 except for situations in which a plus key was pressed. Group 3 contained 5 novices and 2 experts, all of whom gave answers that were identical to Group 1 except for situations in which a number key was pressed. Group 4 contained 2 novices and 4 experts, all of whom gave answers that were similar to those produced by an inexpensive Rockwell model. Group 5 contained one novice and 3 experts, all of whom showed a mixture of acting like Group 1 and Group 4. Finally, there were 7 novices and 6 experts who gave idiosyncratic patterns of answers for the simple problems. Some of the subjects in this category were very similar to one of the above groups except for one or two minor deviations, while others seemed to have highly unique answers. In all cases, however, the answers were internally consistent within subjects as indicated by high correspondence between answers in set A and set B. The answers for each of the common categories on each of the 18 simple problems are given in Table 4.

Models were generated for each of the common categories of answers and for each subject in the miscellaneous category. The production systems for each of the five common categories are given in Tables 5 through 9, respectively.

Insert Tables 5 through 9 about here

Groups 1, 2 and 3 all behave similarly but differ with respect to when an expression is evaluated and displayed. For example, consider the sequence of key strokes: 2 + 3 + 4 = .

According to model 1, subjects think that the calculator evaluates expressions only when a plus key (+) is pressed after a number or when an equals key (=) is pressed. Thus, in the above example, the number in the display after each of the six key strokes will be, 2, 2, 3, 5, 4, 9. The 3 does not get added to the 2 until a plus (or an equals) key is pressed; and the subtotal 5 does not get added to 4 until an equals (or plus) key is pressed.

According to model 2, subjects think that the calculator evaluates expressions only when an equals (=) key is pressed. Thus, in the above example, the numbers in the display after each of the six key strokes will be 2, 2, 3, 3, 4, 9. The entire expression is held in the register until an equals key is pressed.

According to model 3, subjects think that the calculator evaluates expressions as soon as a number key (#) is pressed. Thus, in the above example, the numbers in the display after each respective key stroke will be: 2, 2, 5, 5, 9, 9. The 3 gets added to the 2 as soon as the 3 is pressed; the 4 gets added to the subtotal 5 as soon as the 4 is pressed.

In summary model 2 involves delayed evaluation (i.e., nothing gets evaluated until an equals is pressed), model 3 involves immediate evaluation (i.e., expressions are evaluated as soon as possible), and model 1 involves a compromise between the two extremes (i.e., expressions are evaluated after a plus but not after a number key is pressed). Another way to describe the differences among the first three groups is to say that group 1 treats a plus key like an equals, group 2 treats both a number key and a plus key like an equals, group 3 treats neither like an equals.

The fourth and fifth groups differ from the first three groups with respect to how to deal with the plus key. The fourth group behaves as if the calculator has an automatic constant--evaluation of an expression for a plus or an equals involves adding the number in the display to the number in the register. This mode of evaluation is called "incrementing display" in the tables. The fifth group gives an "incrementing display" only when two plus keys are pressed in sequence. Both groups are like group 1 in that they display the evaluated version of the expression when a plus key or an equals key is pressed but not when a number key is pressed. The "incrementing display" characteristic of group 4, and to some extent group 5, is a more sophisticated and efficient feature of some calculators such as Rockwell.

Insert Table 10 about here

Table 10 gives a list of alternative actions associated with each of the first 6 productions. As can be seen, the five common categories resulted in three alternatives for production P2, three for P4, two for P5, one for P6, two for P7, and three for P8. In the process of developing models for the 14 miscellaneous subjects, several new alternatives were constructed as shown in Table 10. Although a detailed analysis of the performance of each miscellaneous subject would require undue space, examples are given in this section. For example, one of the novices gives the answers 2, 2, 5, 5, 12, 2, 0, 5, 5, 12, 0, 0, 0, 3, 0, 0, 0, 3 for the 18 problems listed in Table 4, respectively. This subject seems to evaluate expressions immediately when a number key is pressed, corresponding to subjects in group 3. However, in addition, this subject treats any irregular sequence of button presses (such as + after +, or + after =) as a reset or clearing of the display and register. The productions which describe this subject's performance are: P2B, P4B, P5C, P6D, P7B and P8B. As another example, one of the experts gives the answers, 2, 2, 3, 5, 7, 2, 4, 5, 10, 12, 2, 4, 8, 3, 4, 8, 16, 7, for the 18 simple problems, respectively. This performance is similar to model 1 except that the display is incremented for = after +. The productions are P2A, P4A, P5A, P6A, P7A; P8C. Models were fit to each of the miscellaneous subjects by taking the best fitting common model (i.e., models 1 through 5) and changing as few productions as necessary in order to make the fit perfect.

The left side of the Table 10 lists the frequencies of each production for the experts and novices. There is a tendency for experts to rely on productions which involve incrementing the display, i.e., P5B and P8C. For example, these productions are used 6 times by novices and 14 times by experts. In addition, there is a tendency for experts to rely on productions which evaluate and display for + and = but not for #, i.e., P2A, P4A, P7A, while novices tend to favor immediate evaluation and display for #, i.e., P2B, P4B, P7B. For example, the former set of productions is used 76 times by experts and 58 times by novices but the latter set is used 13 times by experts and 31 times by novices.

Analysis of Individual Subjects on Multiplication Problems

The previous section encouraged the idea that it is possible to describe the subject's "model of the calculator" for generating answers to 18 simple problems. These analyses were based on problems containing only six possible conditions. In the present section, we expand our analysis to include 16 additional problems which contain multiplication. These are problems 11 through 26 on Table 1. They provide three new conditions: x after # (i.e., pressing the multiply key after pressing a number key, such as 2 x), # after x (i.e., pressing a number key after pressing a multiply key, such as 2 x 3) and = after x (i.e., pressing an equal key after pressing a multiply key, such as 2 x =). Also, up to this point we have considered only conditions which include two events, but this group of questions also allows us to explore whether subjects use more than two events to determine chains of arithmetic; for example, if a subject evaluated all multiplications before additions, then 2 + 3 x 7 = would yield an answer of 23 but if a subject evaluated chains in order of presentation then the answer is 35. Thus, this analysis will allow us to add three new productions to each subject's model developed in the previous section, and to modify some productions for evaluating chain arithmetic.

For each subject, we assume that the six productions established in the previous analysis still are operating for problems 11 through 26. Thus, the goal of the present section is simply to add three new productions (P10, P11, P12) to the model of each subject. All 16 problems contain the condition x after #; almost all of the problems contain the condition, = after x; problems 15, 17, and 23 provide the condition = after x; and several problems involve chain of + and x operations such as problems 24 and 26.

Table 10 shows the most common possible actions associated with each of the three new conditions (P10, P11, P12) explored in this section. As with the analysis of simple problems, one issue concerns when an expression is evaluated. If an expression is evaluated as soon as a multiplication operation (x) or an equals (=) key is pressed, analogous to Model 1 in the previous analysis, then the productions selected would be: P10B (no evaluate for #), P11A (evaluate for x), and P12A (evaluate for equals). If an expression is evaluated only when an equals key is pressed, analogous to the delayed evaluation in Model 2, then the productions selected would be: P10B (no evaluate for #), P11B (no evaluate for x), P12A (evaluate for =). If an expression is evaluated as soon as a number key is pressed, analogous to the immediate evaluation of Model 3, then the productions selected would be: P10A (evaluate for #), P11B (no evaluate for x), P12B (no evaluate for =). Finally, if subjects used an incrementing display for evaluating expressions and numbers as in Model 4 or 5 in the previous section, then the selected productions would be: P10B (no evaluate for #), P11B (no evaluate for x), and P12C (increment display for =). For purposes of this analysis we will refer to each of these four clusters of three productions as Model 1m, Model 2m, Model 3m, and Model 4m, respectively.

Table 11 shows the answers generated by each of the four multiplication models (i.e., productions P10, P11, and P12). Thirty of the 33 novices and 23 of the 33 experts generated performance on the multiplication problems that was consistent with one of the four models; however, specific answers to some problems could differ from those listed in Table 11 in cases where different systems for evaluating chains of arithmetic or different productions for P2 through P8 were in use. The bottom of Table 11 shows the number of novices and experts who were fit by each of the four models.

Insert Table 11 about here

Model 2m allows for evaluation of a chain of arithmetic. Of the novices, 5 subjects performed the arithmetic in order from left to right (as indicated in Table 11), one subject carried out additions before multiplications (e.g., 2 x 3 + 7 = resulted in an answer of 20), one subject carried out multiplications before additions (e.g., 2 + 3 x 7 = resulted in an answer of 23) and one carried out computations only on the last two entries in the register (e.g., 2 x 3 x 7 = resulted in 21; or 2 x 3 + 7 = resulted in 10). Of the experts, three out of five subjects showing model 2m performed chains from left to right, and two of the five experts performed multiplication before addition in a chain.

There were also three miscellaneous novices and 10 miscellaneous experts. Of the novices, two subjects gave model 1m answers for problems that involved only numbers, multiplication, and/or equals but model 2m answers for problems with numbers, addition, multiplication, and equals. One novice gave model 4m answers for problems with numbers, multiplication and/or equals but model 3m answers when problems involved multiplication, addition, numbers and equals.

Thus, these subjects behaved as if the conditions for actions depended on more than just the last two button presses. No additional productions were constructed to try to fit this performance in Table 11. However, for the 10 unique experts, several additional productions for P10, P11 and P12 were constructed and are listed in Table 10. For example, one subject reset the display to zero for x after # and to no change for = after x. The productions for that subject are P10B, P11C, P12A. Another subject ignored the equal sign when it followed the multiplication sign, as indicated by productions P10B, P11B, P12B. Two other subjects reset the display for = after x giving the productions P10B, P11B, P12C. Thus, many of the experts tend to add new productions for unusual button sequences; the effect of most of the new productions is some sort of "reseting" the display. The frequency of use of each alternative production for P10, P11, and P12 for all subjects is summarized in the left side of Table 10.

Analysis of Individual Subjects on Complex Problems

Finally, the performance of each subject on problems 35 through 44 was analyzed. These problems contain many of the conditions already described in the previous two sections; thus, it was assumed that each subject would use the twelve productions already determined by analyzing the first 34 problems in the test. However, problems 35 through 44 also contain four new conditions: x after =, x after x, + after x, and x after +. Thus, in this section four new productions (P13, P14, P15 and P16) are added to the model of each subject.

Table 10 lists the alternative productions for each of the four new conditions explored in this section. Table 12 gives some typical answers by subjects for problems 35 through 44.

Insert Table 12 about here

Group 1 in Table 12 behave as if they were using productions 13A, 14A, 15A, and 16A along with delayed evaluation of expressions (based on earlier productions). These subjects treat x after = and x after x as if there is no change, and if two consecutive operation keys are pressed (such as x after +) they use only the last operation that was pressed. As shown in the bottom of Table 12, there were ten novices and 15 experts who followed this procedure. There were also six more novices and one more expert in group 1'; these subjects behave as if they use the identical four new productions but they show immediate evaluation of expressions when the number key is pressed. In addition four novices (and no experts in group 2 behave like those in group 1 except that when there are two consecutive operations, the multiply "wins", i.e., for 2x+ the register stores 2x. The productions for this group are P13A, P14A, P15B, P16A. Similarly, two more novices and one expert in group 2' use the same four new productions as group 2 but act as if an expression is evaluated as soon as a number key is pressed. In addition, there were 2 novices and 4 experts in group 3. These subjects treat the four new productions as if they serve to increment the display. This procedure is indicated by the combination of P13A, P14B, P15C, P16C. These are the same subjects who increment the display for similar conditions such as + after + or + after = or = after +.

There were also a large number of unique subjects--nine novices and 12 experts. However, almost all of the subjects are closely related to either model 1 or model 3, with just one production slightly different. For example, one subject is like model 1 except that the display is reset to zero for x after + or = after x. The productions for that subject are P13A, P14A, P15D, P16D. Another subject has the same procedure as subjects in group 2 except that x after x results in the display being multiplied by the register; the productions for this subject are P13A, P14B, P15B, P16A. For a third subject, the display is incremented for = after x and x after = and when two consecutive operations are input the multiply wins; the productions are, P13C, P14B, P15B, P16A. The left portion of Table 10 summarizes the frequencies of each of the alternative productions for all subjects on P13, P14, P15 and P16.

General Conclusions

Characterizing the Differences Among Subjects

The present study suggests that subjects differ with respect to their conceptions of the operation of electronic calculators. The foregoing analyses (summarized in Table 10) provided for a detailed description of the differences among subjects, with each subject being described as a list of productions. However, the goal of this section is to provide a more integrated description of the major differences among subjects. Three basic kinds of differences were observed: (1) How is an expression represented in the register? For example, a series of key strokes such as 2+x3 can be represented as 2 + 3 or 2 x 3 or 0 or something else. (2) When is an expression evaluated? For example, does the calculator evaluate at the earliest possible opportunity such as 2 + 3 resulting in a display of 5, does the calculator wait for an equals to be pressed before an evaluation takes place, or does the calculator compromise between these two extremes? (3) How is an expression evaluated? For example, a chain of arithmetic like 2 + 3 x 7 can be evaluated from left to right (answer is 35) or with multiplication first (answer is 23) or in some other way; furthermore, non-standard sequences such as 2+= can be evaluated by incrementing the display to 4 or by ignoring the plus (display is 2) or by reseting the display to 0. In this section, we explore how the subjects differ with respect to their conceptions of how to represent expressions, when to evaluate, and how to evaluate.

Standard conditions. First, there are some general differences which emerge by investigating differences for standard conditions such as # after +, # after x, + after #, x after #, = after #. These are sequences that follow the standard grammar of arithmetic, and are listed as P2, P10, P4, P11, and P7 in Table 10.

The first issue of how to represent expressions is fairly straightforward for all subjects--symbols are added to the register in order from left to right. For example, the keystrokes 2 + 3 x 7 is represented in exactly that way in the register.

The second issue is when to evaluate the expression. In our analysis we located three distinct approaches to the question of when to evaluate. The compromise method is to evaluate an expression whenever an equals key or an arithmetic operation key is pressed but not when a number key is pressed; the immediate method is to evaluate as soon as a number key is pressed (e.g., for 3 + 5 display shows 8); the delayed method is to evaluate only when an equals key is pressed (e.g., for 3 + 5 the display shows 5). The novices and experts tend to differ with respect to their consensus on when to evaluate. Of the novices 13 tend to opt for compromise evaluation, 13 for delayed evaluation, and 7 for immediate evaluation; for experts there is a much stronger consensus of 24 subjects favoring compromise evaluation with 7 favoring delayed evaluation and 2 favoring immediate. A chi-square test revealed that novices and experts differ significantly with respect to the proportion of subjects favoring compromise evaluation, $\chi^2 = 6.15$; df = 1, $p < .05$.

The third issue concerns how to evaluate an expression. In most cases, subjects overwhelmingly follow the normal rules of arithmetic. However, as noted earlier, when subjects use delayed evaluation they may be confronted with a chain of arithmetic such as 2 + 3 x 7. While the majority of subjects evaluate a chain from left to right (i.e., generating an answer of 35), some experts carry out multiplication before addition (i.e., answer is 23), and some novices use other schemes such as carry out additions before multiplications or carry out only the last computation (i.e., answer is 21).

Non-standard conditions (equals after operation). In the present study we also investigated subjects' interpretations of several non-standard conditions, such as = after + or = after x. These are conditions which violate the grammar that demands a number between the operation symbol and the equals symbol. Table 10 represented these as P8 and P12.

The main issue here is how to represent and evaluate an expression when the last key press was an operation and now an equals is pressed. For production P8, the majority of novices (n = 27) and the majority of experts (n = 25) ignore the last + key that was pressed. Thus, a sequence like 2+= results in a display of 2, or a sequence like 2+3+= results in a display of 5. We call this the "no effect" approach because subjects act as if the plus key had no effect. A second approach is what we call the "incrementing" approach; here subjects create some number to go between the + and the = such as the number in the display. For example, the sequence 2+= results in a display of 4 (i.e., it is treated as 2+2=), or 2+3+= may result in 10 (i.e., it is treated as 5+5=) or 8 (i.e., it is treated as 2+3+3=). There were three novices and 6 experts who opted for the incrementing approach. A third option is what we call the "reset" approach. Here subjects reset the display to some number (such as zero) for any non-standard sequence of key presses. Three novices and two experts used a version of the reset approach. The comparable figures for production P12 were 29 novices and 25 experts favored the no effect approach while 4 novices and 8 experts favored the incrementing approach. Although the proportion of subjects favoring the incrementing approach is twice as high for experts as for novices, chi-square tests failed to indicate that the proportion of incrementing subjects was greater for experts in P8, $\chi^2 = .01$, df = 1, or P12, $\chi^2 = .92$, df = 1.

Non-standard conditions (two consecutive operations). Another type of non-standard condition investigated in this study was two consecutive operations such as + after +, x after x, + after x, or x after +. These are conditions which violate the grammatical demand for a number between any two operation symbols. Table 10 presents these as P5, P14, P15, P16.

The main issue here is how to represent and/or evaluate an expression when the last key presses are two arithmetic operators. The three major options chosen by our subjects correspond to those discussed above: "no effect" involves selecting one of the two operation signs to be included in the register and ignoring the other; for example, the most common version of this approach is to ignore the second operation so that 2++ is represented in the register as 2+ or 2xx is represented as 2x. "Incrementing" involves selecting a number to be inserted between the operator symbols; the most common version of this approach is to insert the number from the display so that 2++ becomes 2+2+ or 2xx becomes 2x2x. "Reset" involves clearing the display such as setting it to zero; for example, 2++ results in 0 being displayed. For production P5 the majority of novices (n = 28) and experts (n = 24) opted for the no effect approach; in addition four novices and nine experts opted for the incrementing approach; and one novice and no experts reset the display. The figures for production P14 are similar: 28 novices and 24 experts opted for no effect; incrementing was opted for by 4 novices and 8 experts; and one novice and one expert opted for the reset approach. The patterns for P15 and P16 are similar--the majority of each novices and experts opt for no effect but a substantial number of experts opt for the incrementing option. In all productions, the proportion of experts who increment is more than twice that of the novices. However, even in the most extreme case, the differences in proportion of "incrementers" between experts and novices fails to reach statistical significance, $\chi^2 = 2.27$, df = 1.

Non-standard conditions (operation after equals). Finally, our test involved two productions P6 and P13, which involve + after = and x after = respectively. The status of the register after an = key is pressed is that it contains a number. Thus, these non-standard conditions are most frequently treated in the same way that + after # or x after # is treated. For P6, 30 of the novices and 32 of the experts use this "no effect" approach of simply adding a plus sign to the register. For P13, 28 novices and 27 experts follow the "no effect" approach of adding a multiply sign to the register. However, a sizable minority of the experts (n = 6) opt for an incrementing approach while a sizable minority of the novices opt for a reset option (n = 4).

Summary. The present study provides new information concerning how humans think about calculators. First, we were able to apply the analytic techniques of cognitive psychology to a real-world domain. This allowed a formal and detailed description of how each subject interpreted what was going on inside the "black box" when a key was pressed. Second, in spite of the fact that all of our subjects were proficient in using a calculator to solve standard computational problems, we observed tremendous individual differences among users in their interpretations of the logic of the calculator's operating system. Thus, in spite of apparent similar performance on standard problems, people differ greatly in their knowledge of how the calculator solves problems.

Experts tended to give more consistent answers, as would be expected; however, they also tended to prefer certain operating characteristics such as evaluating an expression for either an operation key or an equal key (compromise evaluator), and incrementing the display during evaluation with non-standard conditions. Further work is needed to determine whether people with certain sets of intuitions can use their calculators more creatively or can learn a new computer language (such as programmable calculators or BASIC) more efficiently than people with other sets of intuitions. In addition, future work is needed to determine whether intuitions--once they have been diagnosed--can be altered through instruction. It is hoped that the groundwork laid in this study will serve as an incentive for continued work in the development of a theory of computer literacy.

Reference Note

1. Suydam, M. N. State of the art review on calculators: Their use in education. Columbus, Ohio: Calculator Information Center, Report No. 3, 1978.

References

Bell, M. Calculators in elementary schools? Some tentative guidelines and questions based on classroom experience. Arithmetic Teacher, 1976, 23, 502-509.

Brown, J. S. & Burton, R. R. Diagnostic models for procedural bugs in basic mathematical skills. Cognitive Science, 1978, 3, 155-192.

Gawronski, J. D. & Coblentz, D. Calculators and the mathematics curriculum. Arithmetic Teacher, 1976, 23, 510-12.

Groen, G. J. & Parkman, J. M. A chronometric analysis of simple addition. Psychological Review, 1972, 79, 329-343.

Hopkins, E. E. A modest proposal concerning the use of hand held calculators in schools. Arithmetic Teacher, 1976, 23, 657-9.

Larkin, J. Information processing models and science instruction. In J. Lochhead and J. Clement (Eds.), Cognitive process instruction. Philadelphia, Franklin Institute Press, 1979.

Mayer, R. E. A psychology of learning BASIC. Communications of the ACM, 1979, 22, 589-593.

Mayer, P. I. When you use a calculator you have to think! Arithmetic Teacher, 1980, 27, 19-21.

Mullish, H. The complete pocket calculator handbook. New York: Collier, 1976.

National Council of Teachers of Mathematics, Instructional Affairs Committee. Minicalculators in schools. Arithmetic Teacher, 1976, 23, 72-6.

Newell, A. & Simon, H. A. Human problem solving. Englewood Cliffs, N. J.: Prentice-Hall, 1972.

Roberts, D. M. The impact of electronic calculators on educational performance. Review of Educational Research, 1980, 50, 71-98.

Roberts, D. M. & Fabrey, L. J. Effects of calculator method, amount of pre-practice and instructional work set on numerical problem solving. Calculators/Computer Magazine, 1979, 2, 63-70.

Roberts, D. M. & Glynn, S. M. Teaching introductory statistics for psychology: Effects of calculators on performance and motivation. Perceptual and Motor Skills, 1979, 48, 563-569.

Rudnick, J. A. & Krulik, S. The minicalculator: Friend or foe? Arithmetic Teacher, 1976, 23, 654-6.

Scandura, A. M., Lowerre, G. F., Veneski, J. & Scandura, J. M. Using electronic calculators with elementary school children. Educational Technology, 1976, 16, No. 8, 14-18.

Schnur, J. O. & Lang, J. W. Just pushing buttons or learning? A case for minicalculators. Arithmetic Teacher, 1976, 23, 559-63.

Simon, D. P. & Simon, H. A. Individual differences in solving physics problems. In R. Siegler (Ed.), Children's thinking: What develops. Hillsdale, N. J.: Erlbaum, 1978.

Woods, S. S., Resnick, L. B. & Groen, G. J. An experimental test of five process models for subtraction. Journal of Educational Psychology, 1975, 67, 17-21.

Footnotes

This research was supported by grant NIE-G-78-0162 from the National Institute of Education. This project was initiated while the senior author was on sabbatical leave at the Learning Research and Development Center, University of Pittsburgh. We appreciate the comments of Richard Young on an earlier version of this project. Requests for reprints should be sent to Richard E. Mayer or Piraye Bayman, Department of Psychology, University of California, Santa Barbara, CA 93106.

[1] Twelve subjects were from the University of Pittsburgh and 34 subjects were from the University of California, Santa Barbara. However, since there were no systematic differences between these groups in age, sex, GPA, SAT scores, or answers to the experimental test, they have been combined into one larger sample.

[2] We wish to thank Dr. Larry Lichten of the Computer Science Department of the University of California, Santa Barbara, for his help in locating subjects for study 2.

[3] For each problem in which the subject gave inconsistent answers between set A and set B, the answer to the A set was used unless it was inconsistent with related items.

[4] Since the HP-21 uses reverse Polish notation (RPN) it was not used as a model in this study.

Table 1

Frequencies of Answers for 44 Problems

Problem Number	Problem	Study 1 Modal Answer	Study 1 Alternatives	Study 2 Modal Answer	Study 2 Alternatives
1	2	(2)=33		(2)=33	
2	2+	(2)=33		(2)=33	(0)=1
3	2+3	(3)=25	(5)=8	(3)=32	(5)=1
4	2+3+	(5)=21	(3)=12	(5)=23	(3)=9;(0)=1
5	2+3+7	(7)=26	(12)=6;(15)=1	(7)=32	(12)=1
6	2=	(2)=33		(2)=33	
7	2+=	(2)=29	(4)=3;(0)=1	(2)=27	(4)=6
8	2+3=	(5)=33		(5)=33	
9	2+3+=	(5)=30	(10)=3	(5)=26	(10)=5;(3)=1;(8)=1
10	2+3+7=	(12)=29	(10)=2;(11)=1(15)=1	(12)=33	
11	2x	(2)=33		(2)=32	(0)=1
12	2x3	(3)=24	(6)=9	(3)=33	
13	2x3x	(6)=24	(3)=9	(6)=23	(3)=9;(0)=1
14	2x3x7	(7)=24	(42)=9	(7)=32	(6)=1
15	2x=	(2)=29	(4)=4	(12)=25	(4)=7;(0)=1
16	2x3=	(6)=33		(6)=33	

Table 1 (Con't.)

Frequencies of Answers for 44 Problems

Problem Number	Problem	Study 1 Modal Answer	Study 1 Alternatives	Study 2 Modal Answer	Study 2 Alternatives
17	2x3x=	(5)=29	(36)=4	(6)=23	(36)=6;(3)=2;(18)=1;(0)=1
18	2x3x7=	(42)=31	(21)=1;(7)=1	(42)=32	(41)=1
19	2+3x	(5)=22	(3)=11	(5)=18	(3)=13;(0)=1;(6)=1
20	2+3x7	(5)=25	(35)=7;(42)=1	(7)=32	(42)=1
21	2x3+	(6)=24	(3)=9	(6)=25	(3)=7;(0)=1
22	2x3+7	(7)=26	(13)=7	(7)=32	(13)=1
23	2+3x=	(5)=26	(25)=3;(3)=2;(11)=1;(2)=1	(5)=23	(25)=3;(3)=3;(11)=2;(2)=1;(15)=1
24	2+3x7=	(35)=29	(23)=2;(21)=1;(42)=1	(35)=23	(23)=7;(42)=2;(45)=1
25	2x3+=	(6)=29	(12)=3;(3)=1	(6)=26	(12)=5;(3)=1;(18)=1
26	2x3+7=	(13)=29	(42)=2;(10)=1;(20)=1	(13)=31	(11)=1;(42)=1
27	2++	(2)=29	(4)=3;(0)=1	(2)=23	(4)=9;(0)=1
28	2+=+	(2)=27	(4)=4;(0)=2	(2)=25	(4)=8;(0)=1
29	2+=+=+	(2)=26	(8)=3;(0)=2;(4)=1;(6)=1	(2)=23	(8)=5;(4)=2;(6)=2;(0)=1
30	2+=+3	(3)=28	(5)-5	(3)=32	(2)=1
31	2++=	(2)=27	(40=3;(8)=2;(0)=1	(2)=22	(4)=6;(8)=4;(6)=1
32	2+=+=	(2)=28	(8)=2;(4)=1;(6)-1;(0)=1	(2)=24	(8)=5;(4)=3;(6)=1

Table 1 (Con't.)

Frequencies of Answers for 44 Problems

Problem Number	Problem	Study 1 Modal Answer	Study 1 Alternatives	Study 2 Modal Answer	Study 2 Alternatives
33	2+=+=+=	(2)=28	(16)=3;(8)=1;(6)=1	(2)=24	(16)=5;(6)=3;(8)=1
34	2+=+3=	(5)=26	(7)=4;(3)=3	(4)=26	(7)=6;(3)=1
35	2xx	(2)=29	(4)=3;(0)=1	(2)=23	(4)=9;(0)=1
36	2x=x	(2)=28	(4)=4;(0)=1	(2)=24	(4)=8;(0)=1
37	2x=x=x	(2)=27	(16)=3;(4)=1;(8)=1;(0)=1	(2)=21	(16)=5;(8)=4;(4)=2;(0)=1
38	2x=x3	(3)=24	(6)=9	(3)=29	(6)=3;(12)=1
39	2xx=	(2)=26	(4)=4;(8)=1;(0)=1;(16)=1	(2)=20	(4)=7;(16)=3;(8)=2;(0)=1
40	2x=x=	(2)=27	(16)=3;(0)=1;(4)=1;(8)=1	(2)=22	(16)=5;(8)=3;(4)=2;(0)=1
41	2x=x=x=	(2)=27	(256)=3(8)=1;(16)=1;90)=1	(2)=21	(256)=5;(16)=2;(8)=3;(32)=1;(0)=1
42	2x=x3=	(6)=24	(3)=5;(12)=3;(48)=1	(6)=24	(12)=7;(3)=1;(0)=1
43	2x+3=	(5)=16	(6)=13;(7)=2;(8)=1;(3)=1	(5)=21	(7)=6;(6)=3;(e)=2;(3)=1
44	2+x3=	(6)=25	(5)=4;(3)=2;(12)=2	(6)=24	(12)=6;(5)=1;(7)=1;(e)=1

Note. - Number in parentheses indicates answer; number to right of equals indicates frequency.

Table 2

Conditions for Simple Problems

Name	Condition	Example	Description
P1	# after #	2 3	Pressing a number key after pressing a number key.
P2*	# after +	+ 3	Pressing a number key after pressing a plus key.
P3	# after =	= 3	Pressing a number key after pressing a equals key.
P4*	+ after #	2 +	Pressing a plus key after pressing a number key.
P5*	+ after +	+ +	Pressing a plus key after pressing a plus key.
P6*	+ after =	= +	Pressing a plus key after pressing an equals key.
P7*	= after #	3 =	Pressing an equals key after pressing a number key.
P8*	= after +	+ =	Pressing an equals key after pressing a plus key.
P9	= after =	= =	Pressing an equals key after pressing an equals key.

Note.--Asterisk (*) indicates that production is relevant to simple problems. P1, P3 and P9 are not relevant since they do not occur in th· simple problems.

Table 3

Some Actions for Simple Problems

Action	Description
D = D	No change in the display
D = #	Erase the old number from the display. Put the new number in the display.
D = R	Erase the old number from the display. Copy the number from the register into the display.
D = eval(R)	Erase the old number from the display. Put the value for the expression in the register into the display.
D = eval(D+R)	Erase the old number from the display. Replace it with the value for the sum of that number from the display and the value in the register.
R = R	No change in register.
R = #	Retain the present expression in the register. Place a number to the right of the expression in the register.
R = "R+#"	Retain the present expression in the register. Place a number to the right of the expression in the register.
R = eval (R)	Erase the old number or expression from the register. Replace it with the evaluation of the number or expression.
R = eval (D+)	Erase the old number or expression from the register. Replace it with the sum of the number in the display plus the evaluation of the register.
R = eval (R)+	Erase the old expression or number from the register. Replace it with the evaluation of that number or expression, and follow that with a plus.

Table 4

Problems and Answers or 18 Simple Items for Four Groups of Subjects

Problem Number	Problem	Group 1 Answer	Group 2 Answer	Group 3 Answer	Group 4 Answer	Group 5 Answer	Miscellaneous
1	2	2	2	2	2	2	
2	2+	2	2	2	2	2	
3	2+3	3	3	5	3	3	
4	2+3+	5	3	5	5	5	
5	2+3+7	7	7	12	7	7	
6	2=	2	2	2	2	2	
7	2+=	2	2	2	4	2	
8	2+3=	5	5	5	5	5	
9	2+3+=	5	5	5	10	5	
10	2+3+7=	12	12	12	12	12	
27	2++	2	2	2	4	4	
28	2+=+	2	2	2	4	2	
29	2+=+=+	2	2	2	8	2	
30	2+=+3	3	3	5	3	3	
31	2++=	2	2	2	8	4	
32	2+=+=	2	2	2	8	2	
33	2+=+=+=	2	2	2	16	2	
34	2+=+3=	5	5	5	7	5	
Number of Subjects-Study 1		8	10	5	2	1	7
Number of Subjects-Study 2		11	7	2	4	3	6

Table 5

Production System for Model 1 (Compromise Evaluation)

Production Name	Condition	Action		Comments
P2A	If # after +	then Set D = #	Set R = "R + #"	
P4A	If + after #	then Set D = eval (R)	Set R = eval (R) +	
P5A	If + after +	then Set D = D	Set R + = R +	(NO CHANGE)
P6A	If + after =	then Set D = eval (R)	Set R = eval (R)	
P7A	If = after #	then Set D = eval (R)	Set R = eval (R)	
P8A	If = after +	then Set D = eval (R)	Set R = eval (R)	

Note.--Evaluates after + or =. Display is non-incrementing.

Table 6

Production System for Model 2 (Delayed Evaluation)

Production	Condition	Action		Comments
P2A	If # after +	then Set D = #	Set R = "R + #"	(SAME AS MODEL 1)
P4B	If + after #	then Set D = D	Set R = eval (R) +	
P5A	If + after +	then Set D = D	Set R + = R +	(SAME AS MODEL 1)
P6A	If + after =	then Set D = D	Set R = R +	(SAME AS MODEL 1)
P7A	If = after #	then Set D = eval (R)	Set R = eval (R)	(SAME AS MODEL 1)
P8A	If = after +	then Set D = eval (R)	Set R = eval (R)	(SAME AS MODEL 1)

Note.--Evaluates after =. Display is non-incrementing.

Table 7

Production System for Model 3 (Immediate Evaluation)

Production Name	Condition	Action		Comments
P2B	If # after +	then Set D = eval (R +#)	Set R = eval (R+#)	
P4B	If + after #	then Set D = D	Set R = eval (R) +	
P5A	If + after +	then Set D = D	Set R = R +	(SAME AS MODEL 1)
P6A	If + after =	then Set D = D	Set R = R +	(SAME AS MODEL 1)
P7B	If = after #	then Set D = D	Set R = R	(NO CHANGE)
P8B	If = after +	then Set D = D	Set R = R	(NO CHANGE)

Note.--Evaluate after #. Display is non-incrementing.

Table 8

Production System for Model 4 (Incrementing Display)

Production Number	Condition	Action		Comments
P2C	If # after +	then Set D = #	Set R = eval (R + #)	
P4C	If + after #	then Set D = R	Set R = R	
P5B	If + after +	then Set D = eval (D+R)	Set R = eval (D+R)	(INCREMENTING DISPLAY)
P6A	If + after =	then Set D = eval (R)	Set R = eval (R)	(SAME AS MODEL 1)
P7A	If = after #	then Set D = eval (R)	Set R = eval (R)	(SAME AS MODEL 1)
P8C	If = after +	then Set D = eval (D+R)	SEt R = eval (D+R)	(INCREMENTING DISPLAY)

Note.--Evaluate after + or =. Display is incrementing.

Table 9

Production System for Model 5 (Partially Incrementing Display)

Production Name	Condition	Action		Comments
P2A	If # after +	then Set D = #	Set R = "R + #"	(SAME AS MODEL 1)
P4A	If + after #	then Set D = eval (R)	Set R = eval (R) +	(SAME AS MODEL 1)
P5B	If + after +	then Set D = eval (D+R)	Set R = eval (D+R)	(SAME AS MODEL 4)
P6A	If + after =	then Set D = eval (R)	Set R = eval (R)	(SAME AS MODEL 1)
P7A	If = after #	then Set D = eval (R)	Set R = eval (R)	(SAME AS MODEL 1)
P8A	If = after +	then Set D = eval (R)	Set R = eval (R)	(SAME AS MODEL 1)

Note.--Compromise between model 1 and model 4.

Table 10

Frequencies of Productions for All Subjects

Frequency in Study 1	Frequency in Study 2	Production Number	Condition	Action	Description
21	27	P2A	If # after +	then Set D = #, Set R = "R+#"	Delayed evaluation and display
7	2	P2B	If# after +	then Set D = eval (R+#), Set R = eval (R+#)	Immediate evaluation and display
4	4	P2C	If # after +	then Set D = #, Set R = eval (R+#)	Immediate evaluation and delayed display
1	0	P2D	If # after +	then Set D = eval (R+D), Set R = eval (R+D)	Immediate evaluation and display with incrementing evaluation
11	18	P4A	If + after #	then Set D = eval (R), Set R = eval (R) +	Immediate evaluation and display
18	9	P4B	If + after #	then Set D = D, Set R = eval (R) +	Immediate evaluation and delayed display
4	4	P4C	If + after #	then Set D = R, Set R=R+	Delayed evaluation and display
0	2	P4D	If + after #	then Set D = 0, Set R = 0	Reset to zero
28	24	P5A	If + after +	then Set D = D, Set R+ = R+	No change

Table 10 (continued)

Frequency in Study 1	Frequency in Study 2	Production Number	Condition	Action	Description
4	9	P5B	If + after +	then Set D = eval (D+R) Set R = eval (D+R)	Immediate incrementing evaluation and display
1	0	P5C	If + after +	then Set D = 0, Set R = 0	Reset to 0
30	32	P6A	If + after =	then Set D = D, Set R = R+	No change in display, add + to expression in register
1	0	p6B	If + after =	then Set D = D, Set R = eval (D+R)	No change in display, immediate incrementing evaluation
1	0	P6C	If + after =	then Set D = 0, Set R = R+	Reset display to 0, add + to expression in register
1	0	P6D	If + after =	then Set D = 0, Set R = 0	Reset display and register to 0, evaluate sum of constant and value in display
0	1	P6E	If + after =	then Set D = D, Set R = eval (#+D)	No change in display

Table 10 (continued)

Frequency in Study 1	Frequency in Study 2	Production Number	Condition	Action	Description
26	31	P7A	If = after #	then Set D = eval (R), Set R = eval (R)	Immediate evaluation and display
6	2	P7B	If = after #	then Set D = D, Set R = R	No change
1	0	P7C	If = after #	then Set D = R, Set R = R	Display value in register
21	23	P8A	If = after +	then Set D = eval (R), Set R = eval (R)	Immediate evaluation and display
6	2	P8B	If = after +	then Set D = D, Set R = R	No change in display or register
2	5	P8C	If = after +	then Set D = eval (D+R) Set R = eval (D+R)	Immediate incrementing evaluation and display
2	0	P8D	If = after +	then Set D = eval (R), Set R = 0	Display the evaluation of the expression in the register, reset the register to 0
1	1	P8E	If = after +	then Set D = eval (D+#), Set R = R	Display the sum of the value in the display plus a constant, no change in register

Table 10 (continued)

Frequency in Study 1	Frequency in Study 2	Production Number	Condition	Action	Description
1	2	P8F	If = after +	then Set D = D, Set = 0	No change in display, set register to 0
8	6	P10A	If # after x	then Set D = eval (R*#) Set R = eval (R*#)	Immediate evaluation and display
25	27	P10B	If # after x	then Set D = #, Set R = R*#	Delayed evaluation and display
11	15	P11A	If x after #	then Set D = eval (R), Set R = eval (R)*	Immediate evaluation and display
19	16	P11B	If x after #	then Set D = D, Set R = eval (R)*	Immediate evaluation and no change in display
0	2	P11C	If x after #	then Set D = 0, Set R = R*	Delayed evaluation and reset display to 0
3	0		(other mixed)		
19	18	P12A	If = after x	then Set D = eval (R), Set R = eval (R)	Immediate evaluation and display

Table 10 (continued)

Frequency in Study 1	Frequency in Study 2	Production Number	Condition	Action	Description
7	7	P12B	If = after x	then Set D = D, Set R = R	No change
4	5	P12C	If = after x	then Set D = eval (D*R), Set R = eval (D*R)	Immediate incrementing evaluation and display
0	1	P12D	If = after x	then Set D = eval (D*R) Set R = eval (R)	Immediate incrementing display, immediate evaluation for register
0	2	P12E	If = after x	then Set D = eval (R*D), Set R = eval (R)* D	Immediate evaluation and display in the constant increment
3	0	(other mixed)			
28	27	P13A	If x after =	then Set D = D, Set R = D*	Delayed evaluation and display
4	0	P13B	If x after -	then Set D = 0, Set R = 0	Reset to 0
1	3	P13C	If x after =	then Set D = eval (D*R), Set R = eval (D*R)	Immediate incrementing evaluation and display
0	1	P13D	If x after =	then Set D = eval (D*R), Set R = R	Immediate incrementing display, no change in register
0	2	P13E	If x after =	then Set D = R, Set eval (D*R)	Immediate incrementing evaluation, delayed display

Table 10 (continued)

Frequency in Study 1	Frequency in Study 2	Production Number	Condition	Action	Description
28	24	P14A	If x after x	then Set D = D, Set R = R*	Delayed evaluation and display
4	8	P14B	If x after x	then Set D = eval (D*R) Set R = eval (D*R)*	Immediate incrementing evaluation and display
1	1	P14C	If x after x	then Set D = 0, Set R = 0	Reset to 0
0	0	P14D	If x after x	then Set D = D, Set R = D*R	Immediate incrementing evaluation, no change in display
19	25	P15A	If + after x	then Set D = D, Set R = R+	Set register sign from * to +
10	2	P15B	If + after x	then Set D = D, Set R = R*	No change
2	4	P15C	If + after x	then Set D = eval (D*R), Set R = eval (D*R)	Immediate incrementing evaluation and display
2	1	P15D	If + after x	then Set D = 0, Set R = 0	Reset to 0
0	1	P15E	If + after x	then Set D = eval (D*R), Set R = eval (D*R)+	Immediate incrementing evaluation and display with register sign to +

Table 10 (Continued)

Frequency in Study 1	Frequency in Study 2	Production Number	Condition	Action	Description
28	25	P16A	If x after +	then Set D = D, Set R = R*	Set register sign from + to *
1	0	P16B	If x after +	then Set D = D, Set R = R+	No change
2	4	P16C	If x after +	then Set D = eval (D*R), Set R = eval (D*R)	Immediate incrementing evaluation and display
2	1	P16D	If x after +	then Set D = 0, Set R = 0	Reset to 0
0	3	P16E	If x after +	then Set D = eval (D*R), Set R = eval (D*R)*	Immediate incrementing evaluation and display with register sign set to *

Table 11

Problems and Answers for 17 Multipliction Items by Four Groups of Subjects

Problem Number	Problem	Group 1m Answer	Group 2m Answer	Group 3m Answer	Group 4m Answer	Miscellaneous
11	2 x	2	2	2	2	
12	2 x 3	3	3	6	3	
13	2 x 3 x	6	3	6	6	
14	2 x 3 x 7	7	7	42	7	
15	2 x =	2	2	2	4	
16	2 x 3 =	6	6	6	6	
17	2 x 3 x =	6	6	6	36	
18	2 x 3 x 7 =	42	42	42	42	
19	2 + 3 x	5	3	5	5	
20	2 + 3 x 7	7	7	35	7	
21	2 x 3 +	6	3	6	6	
22	2 x 3 + 7	7	7	13	7	
23	2 + 3 x =	5	5	5	25	
24	2 + 3 x 7 =	35	35	35	35	
25	2 x 3 + =	6	6	6	12	
26	2 x 3 + 7 =	13	13	13	13	
Number of Subjects-Study 1		11	8	7	5	3
Number of Subjects-Study 2		15	5	0	3	10

Table 12

Problems and Answers for 10 Complex Items by Five Groups of Subjects

Problem Number	Problem	Group 1 Answer	Group 1' Answer	Group 2 Answer	Group 2' Answer	Group 3 Answer	Miscellaneous
35	2 x x	2	2	2	2	4	
36	2 x = x	2	2	2	2	4	
37	2 x = x = x	2	2	2	2	16	
38	3 x = x = 3	3	6	3	6	3	
39	3 x x =	2	2	2	2	16	
40	2 x = x =	2	2	2	2	16	
41	2 x = x = x =	2	2	2	2	256	
42	2 x = x 3 =	6	6	6	6	12	
43	2 x + 3 =	5	5	6	6	7	
44	2 + x 3 =	6	6	6	6	12	
Number of Subjects-Study 1		10	6	4	2	2	9
Number of Subjects-Study 2		15	1	0	1	4	12

TECHNICAL REPORT SERIES IN LEARNING AND COGNITION

Report No.	Authors and Title
78-1	Peper, R.J. & Mayer, R.E. Note-taking as a Generative Activity. (Journal of Educational Psychology, 1978, 70, 514-522.)
78-2	Mayer, R.E. & Bromage, B. Different Recall Protocols for Technical Text due to Advance Organizers. (Journal of Educational Psychology, 1980, 72, 209-225.)
79-1	Mayer, R.E. Twenty Years of Research on Advance Organizers. (Instructional Science, 1979, 8, 133-167.)
79-2	Mayer, R.E. Analysis of a Simple Computer Programming Language: Transactions, Prestatements and Chunks. (Communications of the ACM, 1979, 22, 589-593.)
79-3	Mayer, R.E. Elaboration techniques for Technical Text: An Experimental Test of the Learning Strategy Hypothesis. (Journal of Educational Psychology, 1980, 72, in press.)
80-1	Mayer, R.E. Cognitive Psychology and Mathematical Problem Solving. (Proceedings of the 4th International Congress on Mathematical Education, 1980.)
80-2	Mayer, R.E. Different Solution Procedures for Algebra Word and Equation Problems.
80-3	Mayer, R.E. Schemas for Algebra Story Problems.
80-4	Mayer, R.E. & Bayman, P. Analysis of Users' Intuitions About the Operation of Electronic Calculators.
80-5	Mayer, R.E. Recall of Algebra Story Problems.
80-6	Mayer, R.E. Solving Nonsense and Meaningful Equations.